Setting cohesive elements at all interfaces between elements in an arbitrary mesh

Using ciGEN code by Dr. V.P. Nguyen

Step 1: generating geometry using gmsh

Definition of points, lines, circles, etc. (in .geo file)

```
// Square geometry with cylindrical hole

Point(1) = {0, 0, 0, 1.0};
Point(2) = {30, 0, 0, 1.0};
Point(3) = {30, 30, 0, 1.0};
Point(4) = {0, 30, 0, 1.0};
```

```
Line(1) = {1, 10};
Line(2) = {10, 2};
Line(3) = {2, 11};
Line(4) = {11, 3};
Line(5) = {3, 12};
Line(6) = {12, 4};
Line(7) = {4, 13};
Line(8) = {13, 1};
```

```
// start point, centre, end point
Circle(13) = {6, 5, 7};
Circle(14) = {7, 5, 8};
Circle(15) = {8, 5, 9};
Circle(16) = {9, 5, 6};
```

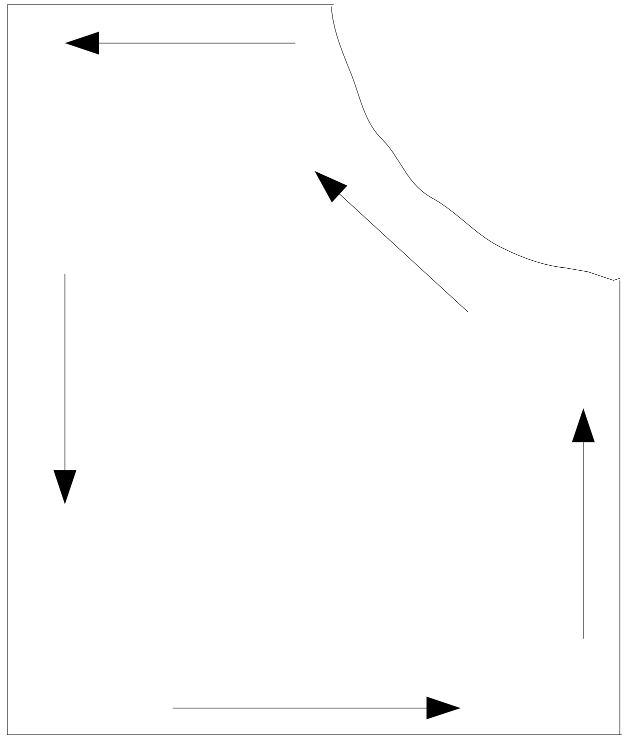
Step 1: generating geometry using gmsh

Definition of line loops and plane surfaces

```
// Line loops

Line Loop(101) = {1, 9, -16, -12, 8};
Line Loop(102) = {2, 3, 10, -13, -9};
Line Loop(103) = {4, 5, 11, -14, -10};
Line Loop(104) = {6, 7, 12, -15, -11};

Plane Surface(201) = {101};
Plane Surface(202) = {102};
Plane Surface(203) = {103};
Plane Surface(204) = {104};
```



Step 1: generating mesh using gmsh

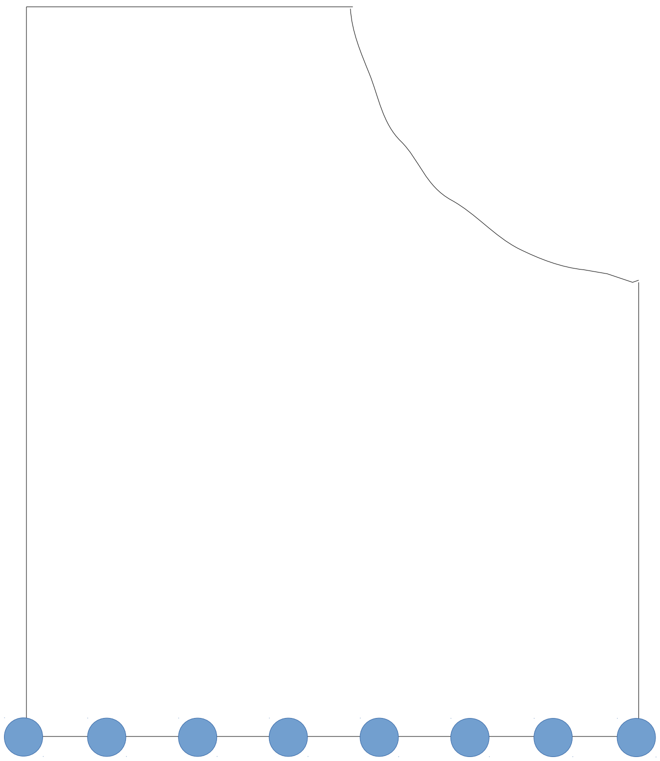
Transfinite/recombine algorithm to define mesh seeds on the edges of the geometry

```
// Transfinite Line

Transfinite Line {1, 2, 3, 4, 5, 6, 7, 8} = 31 Using Progression 1;
Transfinite Line {9, 10, 11, 12} = 11 Using Progression 1;
Transfinite Line {13, 14, 15, 16} = 9 Using Progression 1;

// Recombine Surface

Recombine Surface {201};
Recombine Surface {202};
Recombine Surface {203};
Recombine Surface {204};
```



Step 1: generating mesh using gmsh

Extrusion of the 2D geometry to create a 3D geometry

```
Extrude {0, 0, 1.0} {
  Surface{201,202,203,204};
  Layers{1};
  Recombine;
}

Physical Volume("Part1") = {1,2,3,4};
```

➡ Definition of the physical volume is necessary for ciGEN to identify the volume where element connectivity is parsed

Step 1: generating mesh using gmsh

Definition of physical surfaces will help to generate boundary conditions in Abaqus because when mesh is imported, Abaqus does not recognise a surface entity, meaning cannot be selected easily

```
Physical Surface("Top") = {26};
Physical Surface("Bottom") = {1};
Physical Surface("Right") = {21};
Physical Surface("Left") = {13};
Physical Surface("Front") = {17};
Physical Surface("Back") = {25};
```

Step 2: download and setup ciGEN

Download from github

vinhphunguyen/ciGEN

https://github.com/vinhphunguyen/ciGEN/tree/203ec37233fd739a67b66ca911311c8292f747c4

Necessary installations:

sudo apt-get install libboost-dev
sudo apt-get install scons

Compile ciGEN using “scons” or “make all” command inside the src folder

Procedure tested in Ubuntu 18

Step 3: generate mesh and execute ciGEN

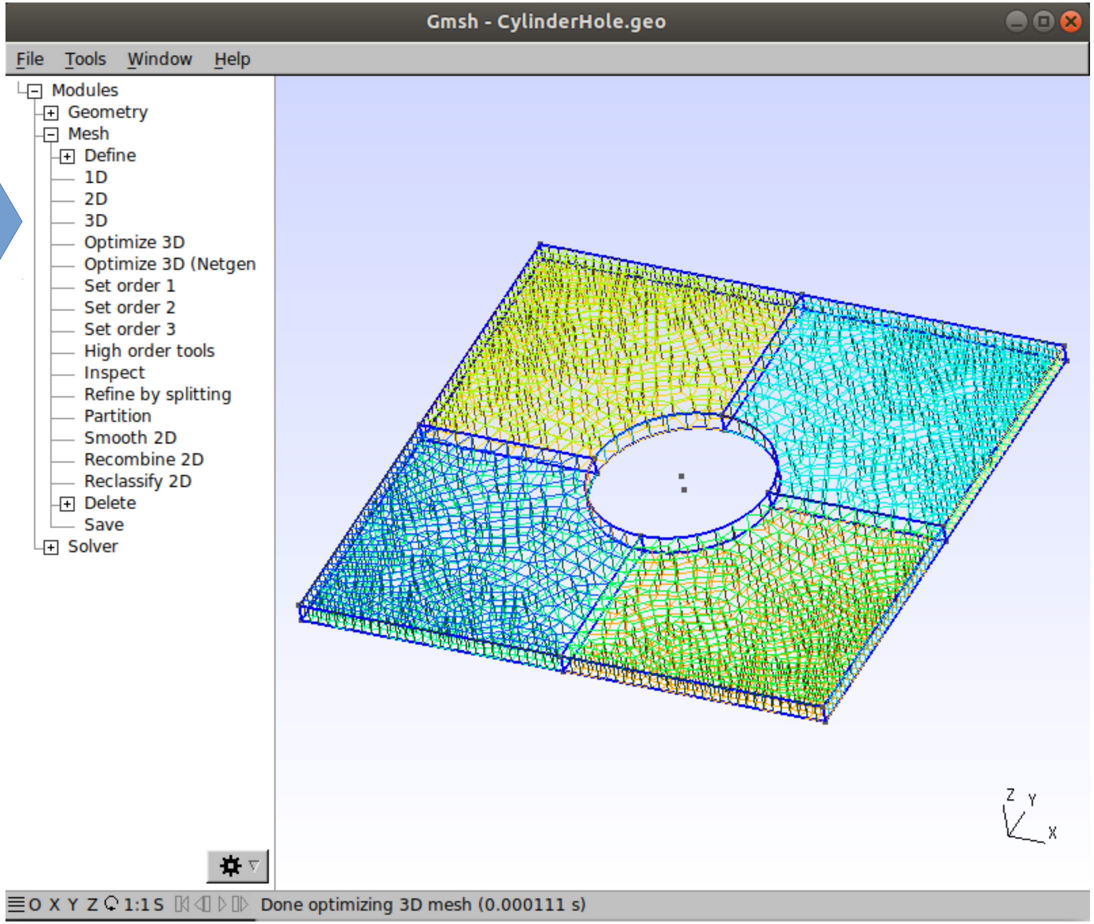
Move .geo file (gmsh file) in ciGEN/src

Run: “gmsh filename.geo”

Define 3D mesh

File > Save Mesh (Ctrl+Shift+S)

It will generate the .msh file



Step 3: generate mesh and execute ciGEN

.msh file contains nodes and elements

```
$MeshFormat
2.2 0 8
$EndMeshFormat
$PhysicalNames
1
3 1 "Part1"
$EndPhysicalNames
$Nodes
4956
1 0 0 0
2 30 0 0
3 30 30 0
4 0 30 0
5 15 10 0
6 20 15 0
7 15 20 0
```

Physical names section must be present in order for ciGEN to work

```
$Elements
2342
1 5 2 1 1 768 787 655 1043 2938 2957 2825 3213
2 5 2 1 1 650 967 765 835 2820 3137 2935 3005
3 5 2 1 1 893 1150 750 682 3063 3320 2920 2852
4 5 2 1 1 772 932 733 1057 2942 3102 2903 3227
5 5 2 1 1 920 827 264 265 3090 2997 357 358
6 5 2 1 1 962 744 988 628 3132 2914 3158 2798
7 5 2 1 1 680 865 712 901 2850 3035 2882 3071
8 5 2 1 1 286 285 799 738 372 373 2969 2908
9 5 2 1 1 879 672 953 747 3049 2842 3123 2917
```

Step 3: generate mesh and execute ciGEN

Execute ciGEN on the .msh file (gmsh file) in ciGEN/src

Run:

./mesh-generator --everywhere --mesh-file CylinderHole.msh
--out-file CylinderHole.inp

.inp extension tells to the generator that Abaqus files are requested

Step 4: setup input file

Files generated are:

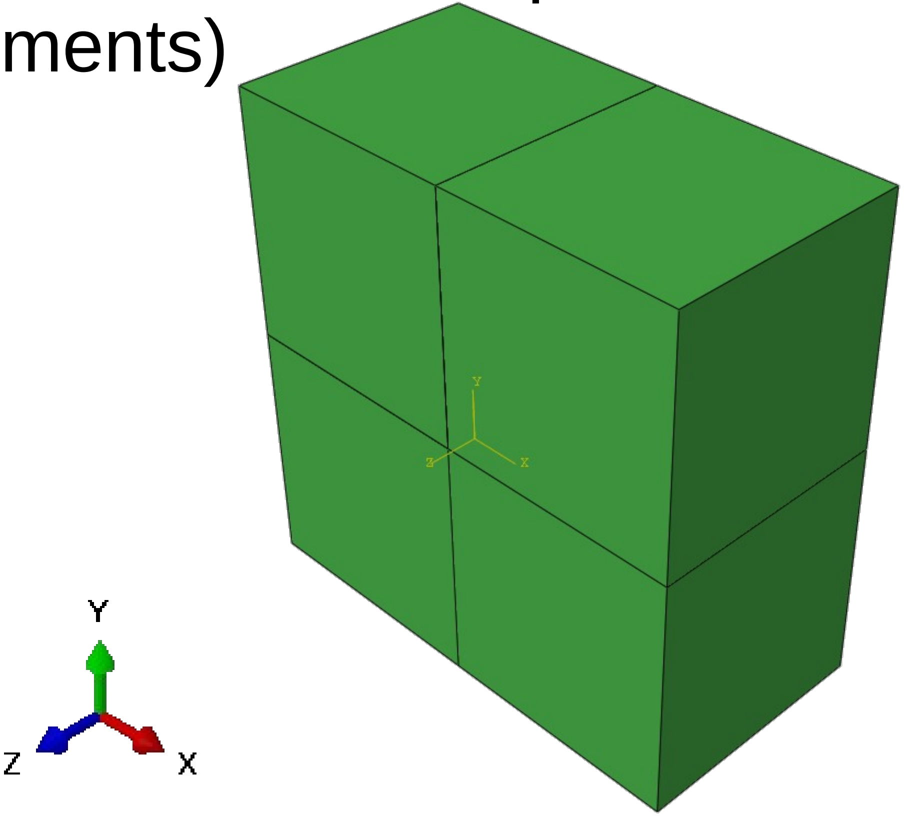
CylinderHole.inp
CylinderHole-node.inp → nodes
CylinderHole-bulk-elems.inp → Bulk elements
CylinderHole-int-elems.inp → Interface elements

Create an Abaqus input file with only nodes and elements, set the type of bulk elements (e.g. type=C3D8). Abaqus CAE will not allow to import user elements (in CylinderHole-int-elems.inp file), these will be added later

Step 4: setup input file

Abaqus CAE: File > Import > Model

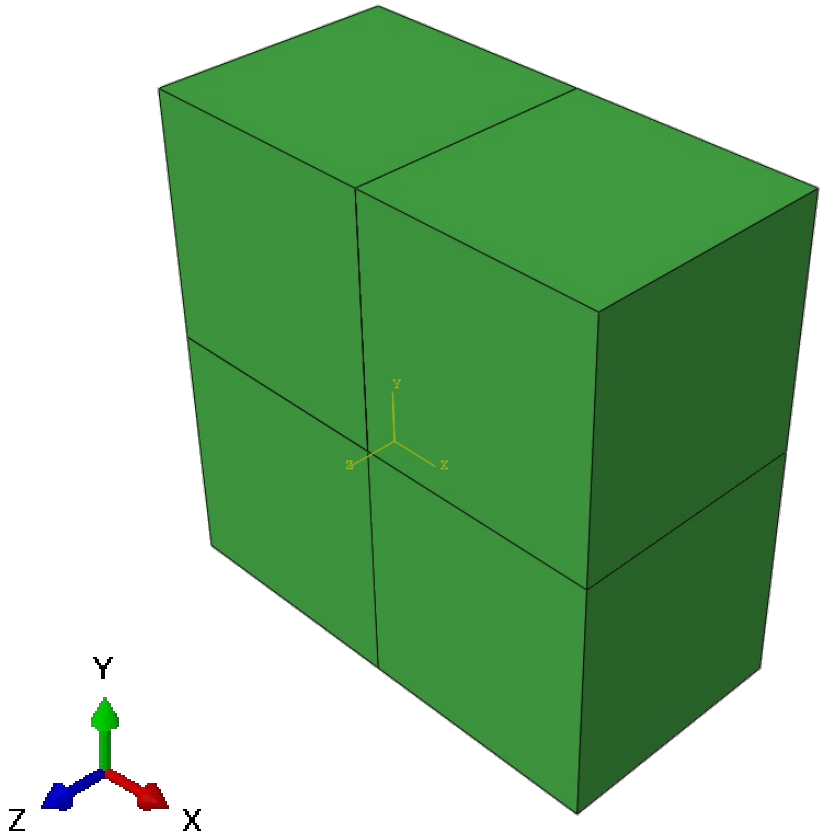
Choose file with .inp extension (with only nodes and bulk elements)



At this point setup boundary conditions, steps, output, as a standard Abaqus analysis

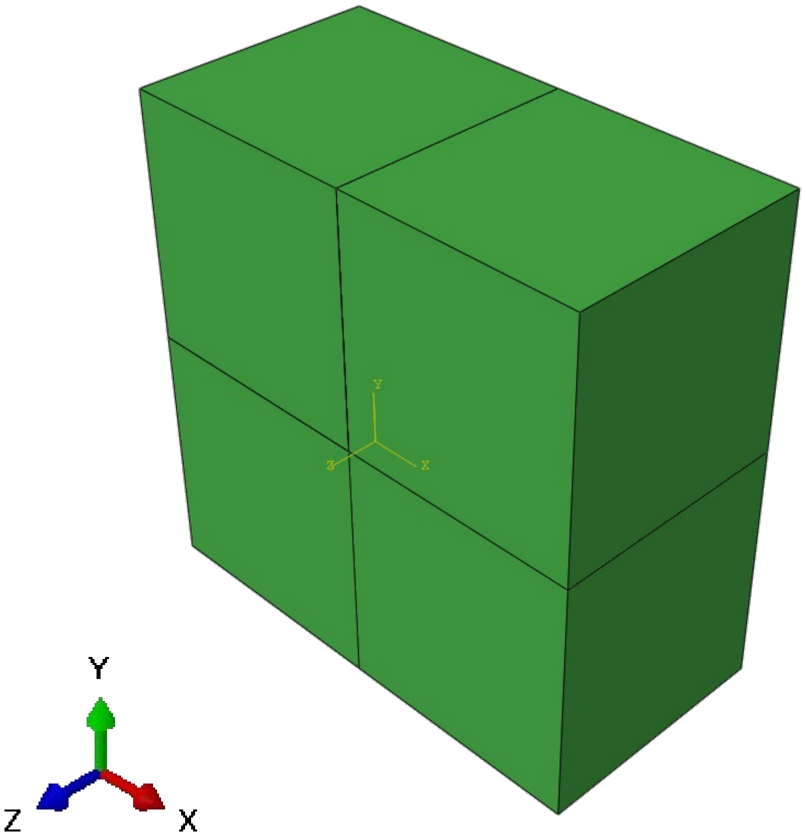
Generate input file (will not contain interface elements because they have been removed)

Nodes: note they are already duplicated, that allows definition of interface elements



18, 0.5, 0.5, 0.5
19, 0.5, 0, 0
20, 1, 0.5, 0
21, 0, 0.5, 0
22, 0.5, 1, 0
23, 0, 0.5, 0.5
24, 0.5, 0, 0.5
25, 1, 0.5, 0.5
26, 0.5, 1, 0.5
27, 0.5, 0.5, 0
28, 0.5, 0.5, 0
29, 0.5, 0.5, 0
30, 0.5, 0.5, 0.5
31, 0.5, 0.5, 0.5
32, 0.5, 0.5, 0.5

Elements: note duplicated nodes belong to different elements



*ELEMENT, TYPE=C3D8
1, 4, 11, 17, 12, 5, 13, 18, 16
2, 22, 27, 10, 3, 26, 30, 15, 8
3, 21, 1, 9, 28, 23, 6, 14, 31
4, 29, 19, 2, 20, 32, 24, 7, 25

Step 5: add interface elements and user element properties

Just after the bulk elements add:

```
*USER ELEMENT, TYPE=U1, NODE=8, COORDINATES=3, PROPERTIES=11, VARIABLES=24
1, 2, 3
```

3D analysis

Specific for the particular UEL used

Cohesive user elements with 8 nodes

3x8

Step 5: add interface elements and user element properties

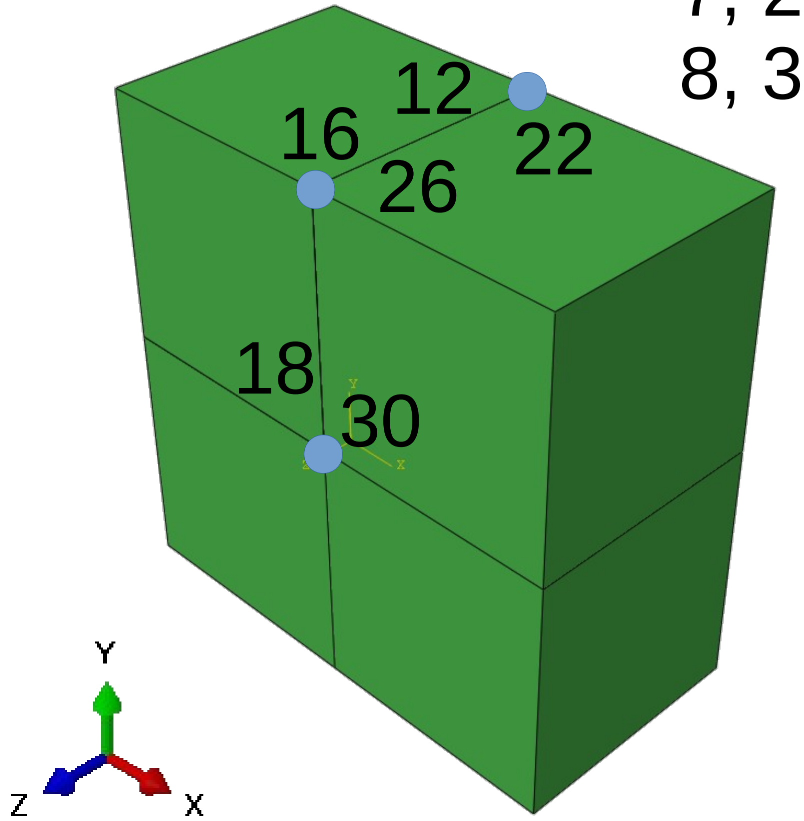
Just after the previous user element definition, add the interface elements from file filename-int-elems.inp

```
*ELEMENT, TYPE=U1, ELSET=COHELE
5, 16, 18, 17, 12, 26, 30, 27, 22
6, 11, 13, 18, 17, 21, 23, 31, 28
7, 27, 30, 15, 10, 29, 32, 25, 20
8, 31, 14, 9, 28, 32, 24, 19, 29
```

Name the set

Interface elements:

```
*ELEMENT, TYPE=U1, ELSET=COHELE
5, 16, 18, 17, 12, 26, 30, 27, 22
6, 11, 13, 18, 17, 21, 23, 31, 28
7, 27, 30, 15, 10, 29, 32, 25, 20
8, 31, 14, 9, 28, 32, 24, 19, 29
```



Step 5: add interface elements and user element properties

Add UEL properties and interface element set name just after the section definition (section used for bulk elements)

```
** Section: LinearElastic
*Solid Section, elset=LinearElastic, material=LinearElastic
,
*UEL PROPERTY, ELSET=COHELE
0.5, 1.0, 20, 300, 3, 3, 0.02, 0.02
1, 1, 1
```

In this case bulk elements are just linear elastic and UEL used is the one from Georgia Tech, Glaucio H. Paulino

Step 6: download UEL and put it in simulation folder

https://paulino.ce.gatech.edu/PPR_tutorial.html

Fortran UEL and Resources

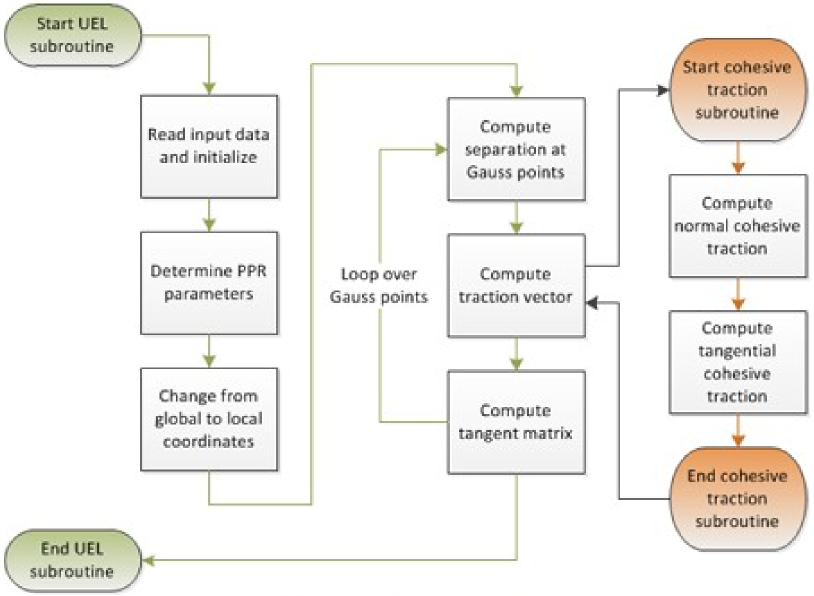
The flowchart to the left shows the major steps of the PPR UEL, which is available below. The subroutine is entered and exited from the ABAQUS finite element routine at the green ellipses.

Downloads:

2D UEL (zip format)
3D BRICK UEL (zip format)
3D TET4 UEL (zip format)
3D TET10 UEL (3D) (zip format)

Additional PPR Resources:

- **Instructions for implementing the PPR cohesive model in ABAQUS**
- **Nomenclature** (equivalence between variables in paper and in UEL)



Flow chart of the PPR UEL

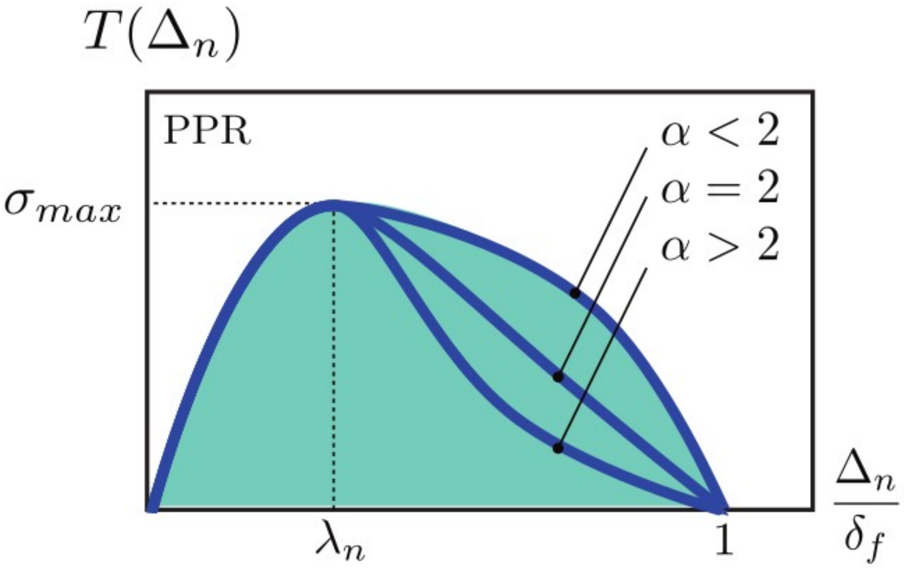
Download “3D BRICK UEL”, extract and put it in simulation folder with input file, modify to .for for Windows

Cohesive law

$$T(\Delta_n) = \begin{cases} \frac{\phi_n}{\delta_n}\left(\frac{\alpha}{m}\right)^m \left(1-\frac{\Delta_n}{\delta_n}\right)^{\alpha-1}\left(\frac{m}{\alpha}+\frac{\Delta_n}{\delta_n}\right)^{m-1}(\alpha+m)\frac{\Delta_n}{\delta_n}, & \Delta_n \leqslant \delta_f, \\ 0, & \Delta_n > \delta_f, \end{cases}$$

$$\delta_n = \frac{\phi_n}{\sigma_{max}}\alpha\lambda_n(1-\lambda_n)^{\alpha-1}\left(\frac{\alpha}{m}+1\right)\left(\frac{\alpha}{m}\lambda_n+1\right)^{m-1}$$

$$m = \frac{\alpha(\alpha-1)\lambda_n^2}{(1-\alpha\lambda_n^2)}$$



*UEL PROPERTY, ELSET=COHELE
0.5, 1.0, 20, 300, 3, 3, 0.02, 0.02
1, 1, 1

Max stress (normal)

Max stress (tangential)

Cohesive law

*UEL PROPERTY, ELSET=COHELE
0.5, 1.0, 20, 300, 3, 3, 0.02, 0.02
1, 1, 1

Separation at failure (normal and tangential)

*UEL PROPERTY, ELSET=COHELE
0.5, 1.0, 20, 300, 3, 3, 0.02, 0.02
1, 1, 1

α

Fracture energy

Step 7: run the simulation

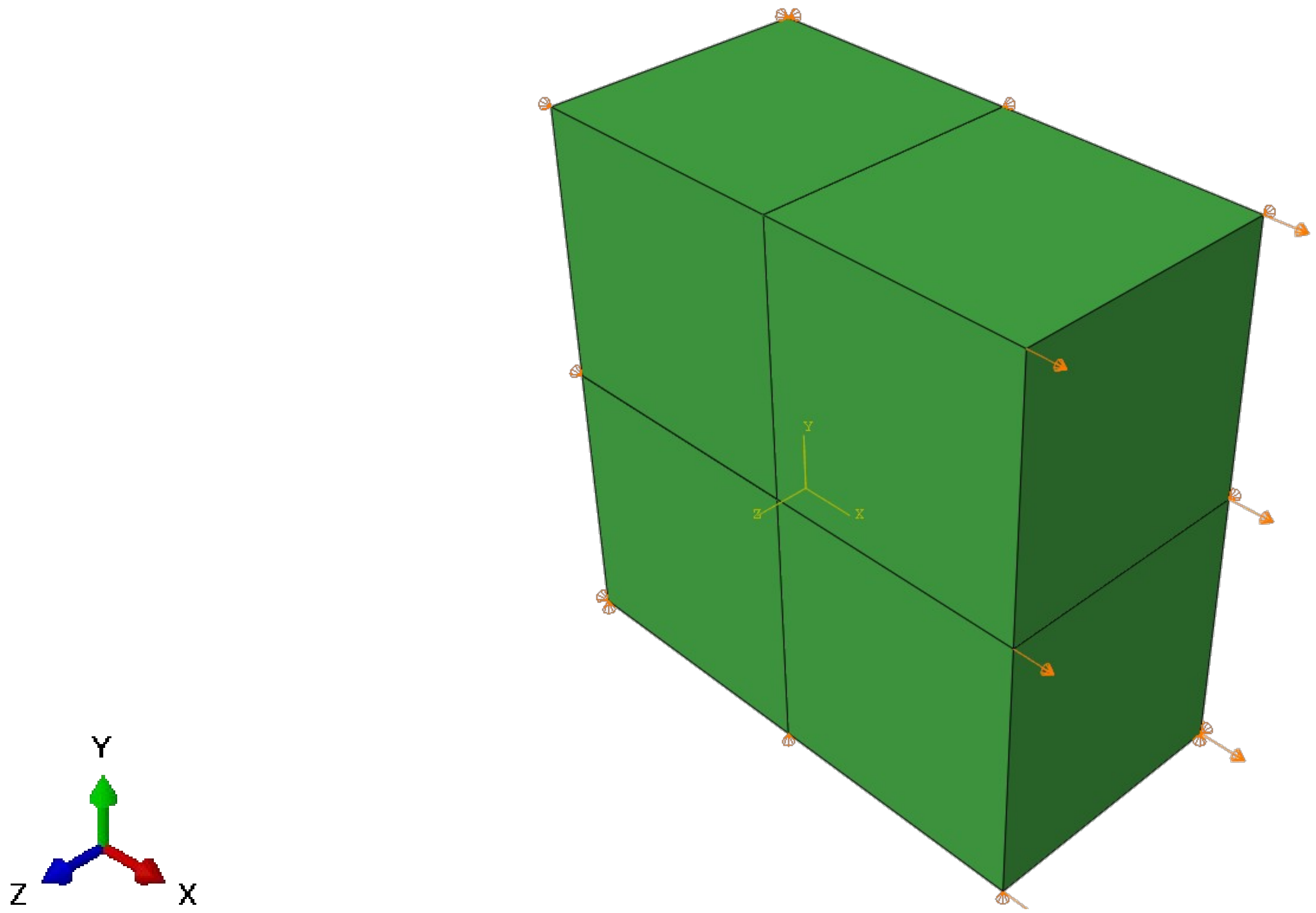
Set Riks solver to obtain convergence after max stress

```
*Step, name=Tension, nlgeom=YES, inc=10000
*Static, riks
0.0005, 1., 1e-06, 0.0005
```

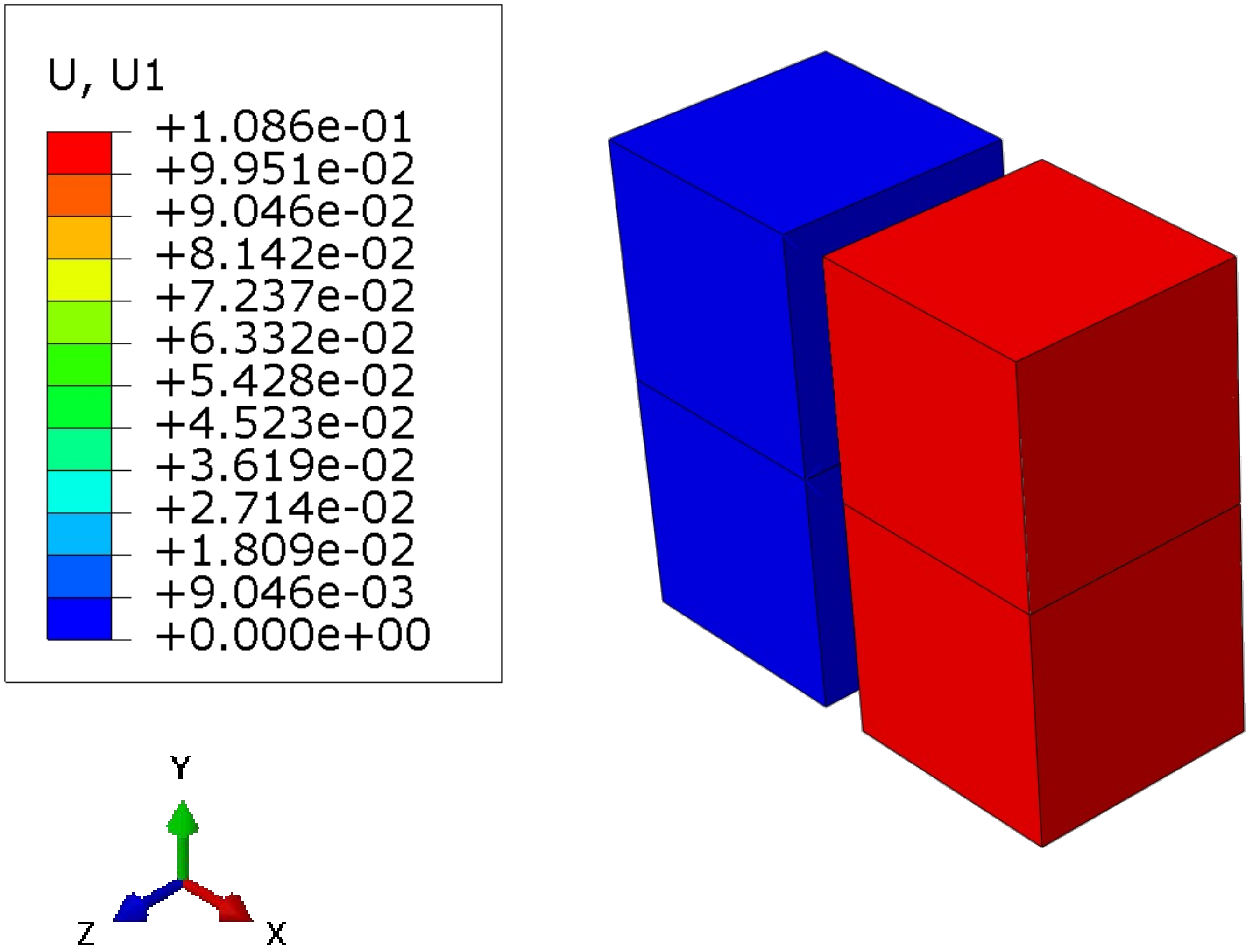
Run in the simulation folder:

```
abaqus job=Job-1 user=3DpprBRICK.for
```

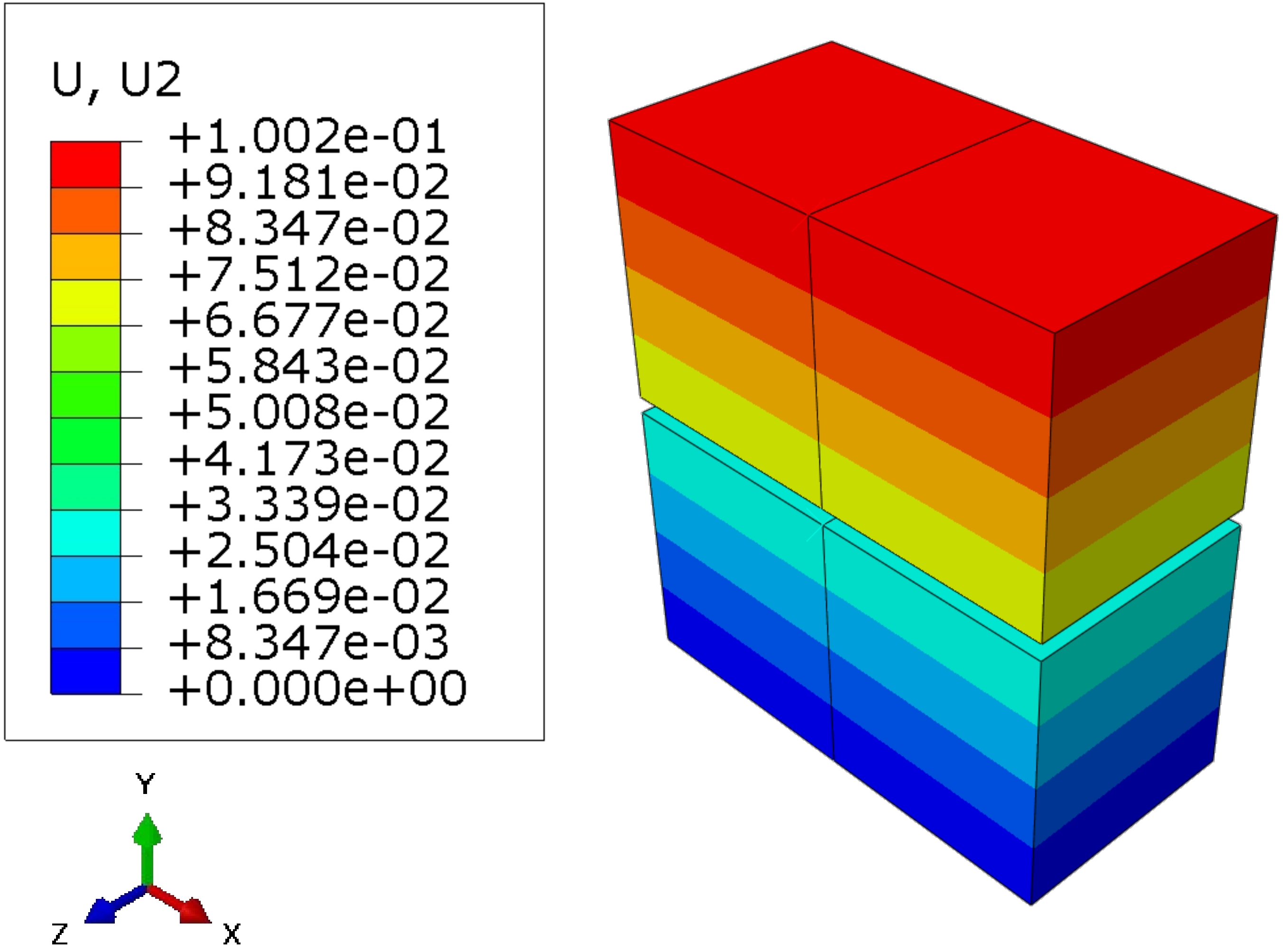
Test: cases 1) Tension along y, 2) Tension along x



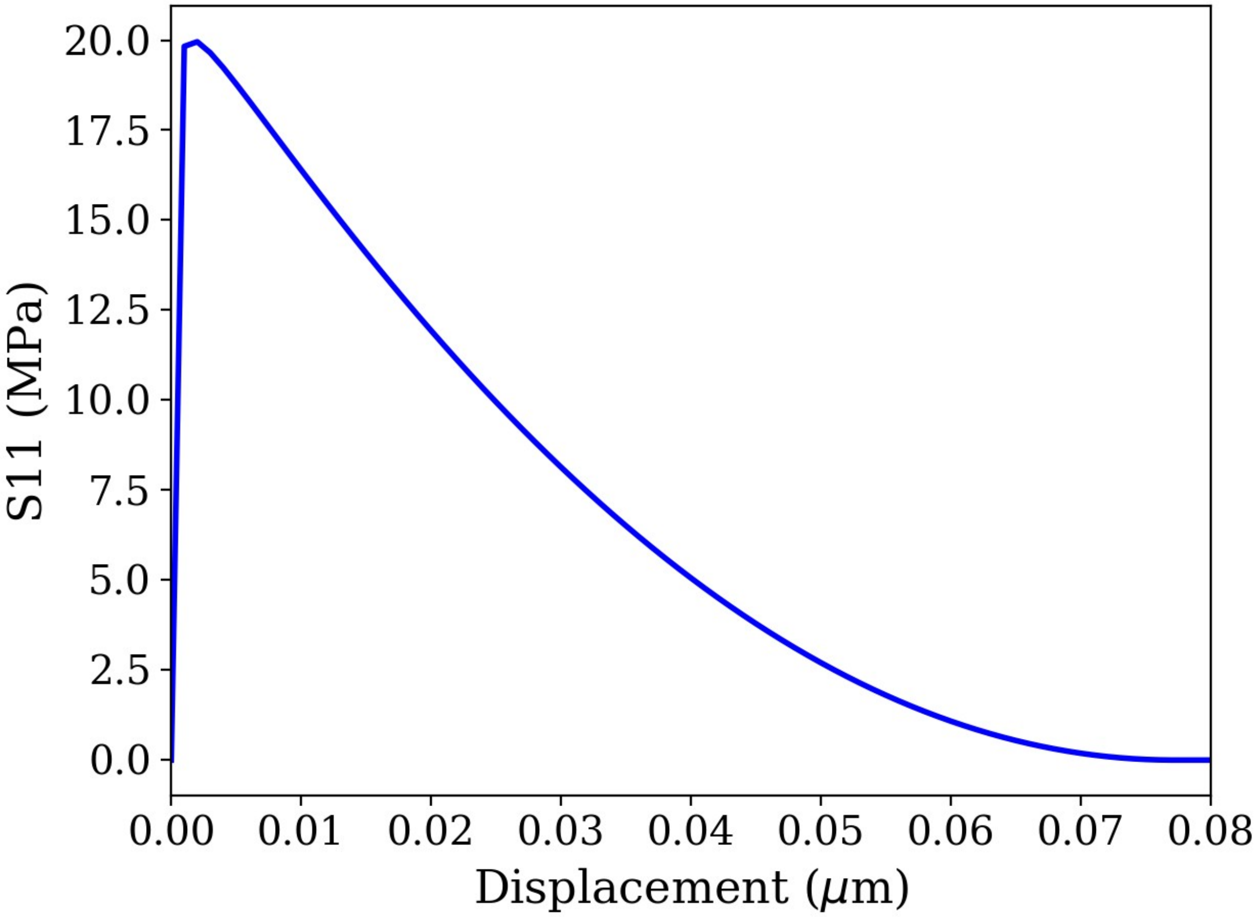
Tension along x



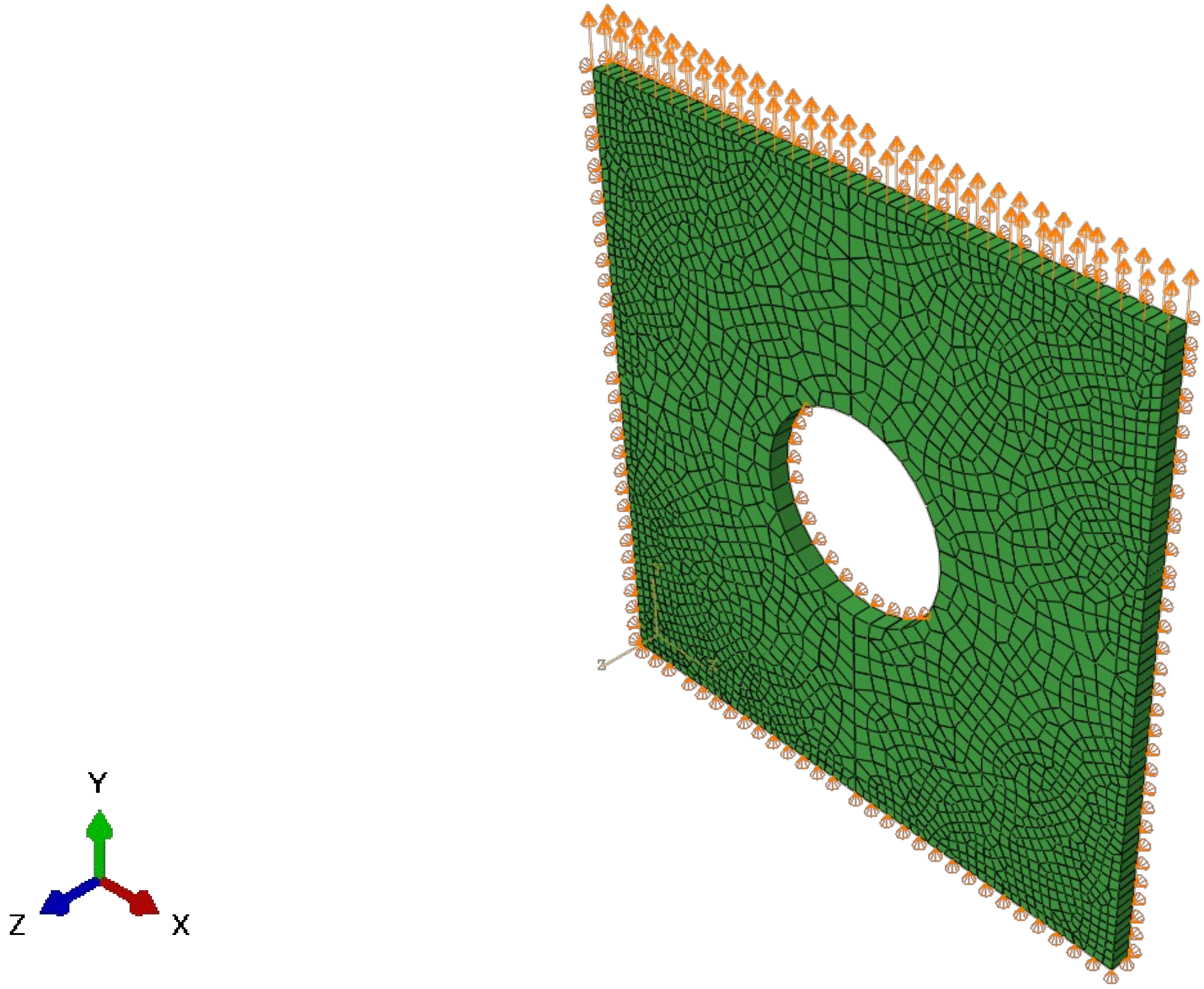
Tension along y



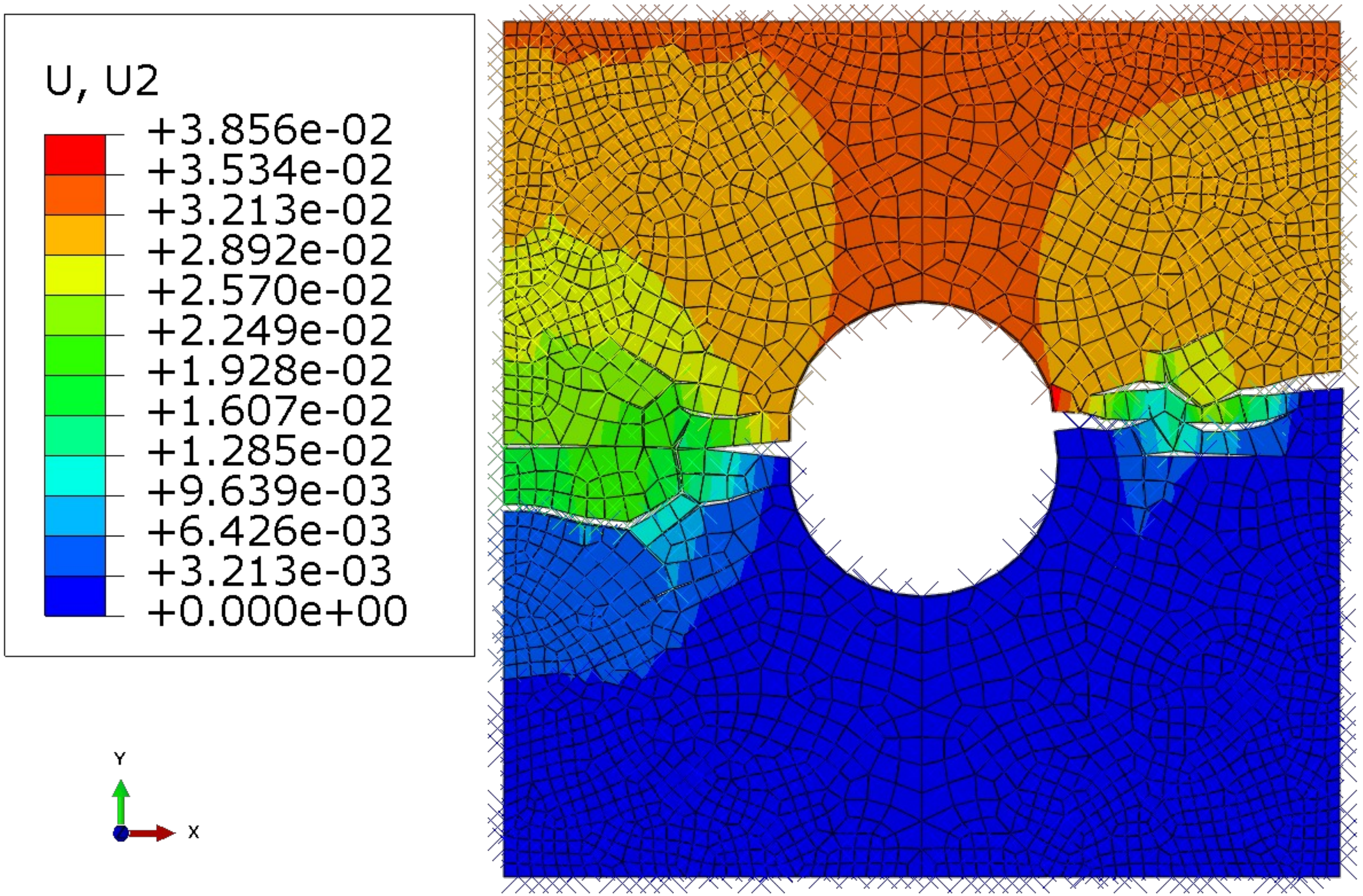
Stress-strain curve



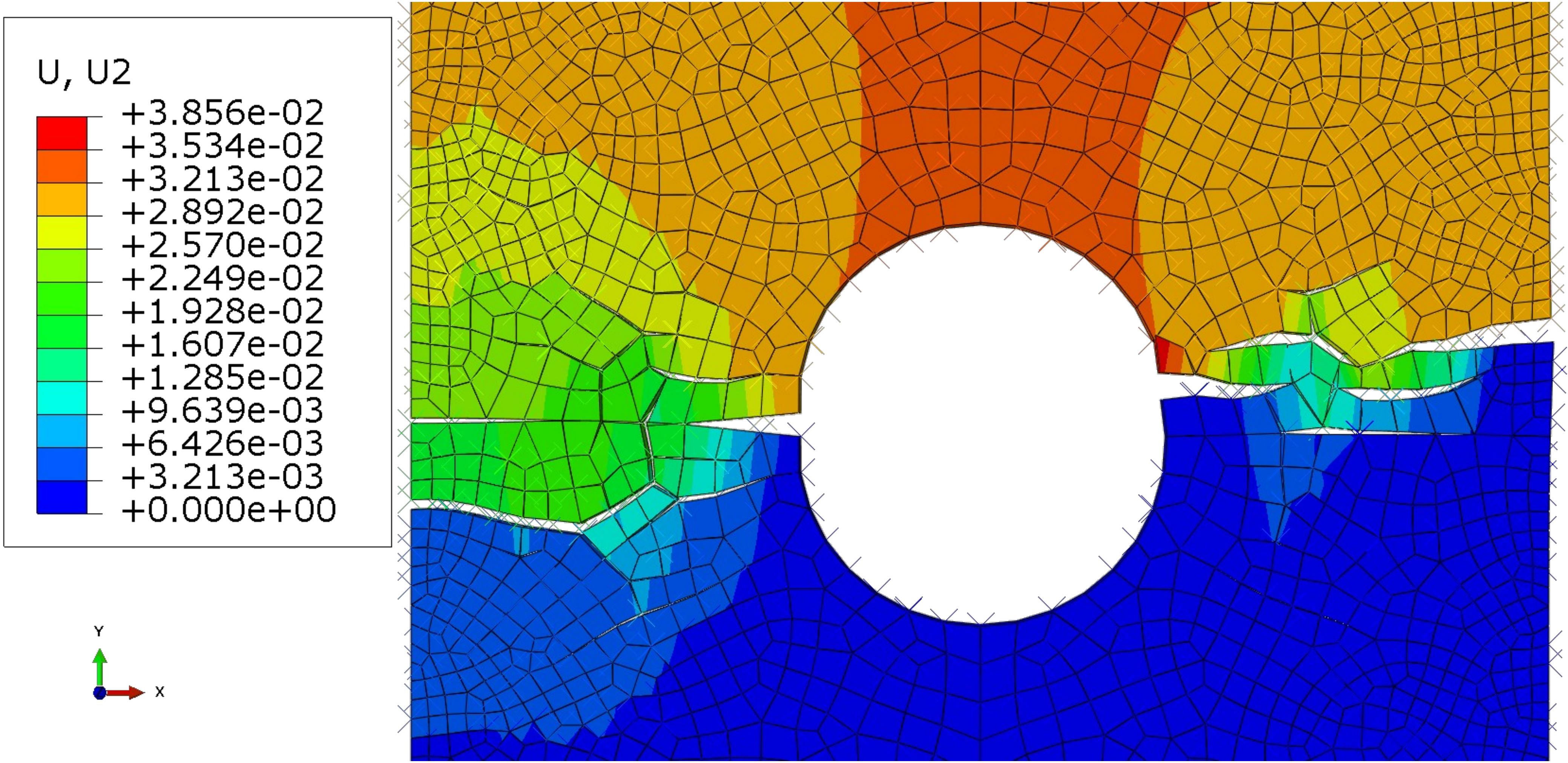
Test with arbitrary mesh



Displacement field and fracture



Displacement field and fracture



Conclusions

1) Possible to insert cohesive elements everywhere in the geometry for an arbitrary mesh

Outlook

2) Need to test the convergence with linear cohesive law UEL or Abaqus built-in cohesive elements

3) Coupling with crystal plasticity model